
UNIVERSITI SAINS MALAYSIA

Second Semester Examination
2010/2011 Academic Session

April/May 2011

CPT212 – Design & Analysis of Algorithms
[Reka Bentuk & Analisis Algoritma]

Duration : 2 hours
[Masa : 2 jam]

INSTRUCTIONS TO CANDIDATE:
[ARAHAN KEPADA CALON:]

- Please ensure that this examination paper contains **FIVE** questions in **NINE** printed pages before you begin the examination.

*[Sila pastikan bahawa kertas peperiksaan ini mengandungi **LIMA** soalan di dalam **SEMBILAN** muka surat yang bercetak sebelum anda memulakan peperiksaan ini.]*

- Answer **any FOUR** questions only.

*[Jawab **mana-mana EMPAT** soalan sahaja.]*

- You may answer the questions either in English or in bahasa Malaysia.

[Anda dibenarkan menjawab soalan sama ada dalam bahasa Inggeris atau bahasa Malaysia.]

- In the event of any discrepancies, the English version shall be used.

[Sekiranya terdapat sebarang percanggahan pada soalan peperiksaan, versi bahasa Inggeris hendaklah diguna pakai.]

1. (a) (i) Write an algorithm in Java/C++ that finds the minimum and maximum values in an array of integer values, using the methods below.

- (A) using a **for** loop
(B) recursive method without using any loop

Tulis satu algoritma dalam Java/C++ yang mencari nilai minimum dan maksimum dalam satu tatasusunan integer, menggunakan kaedah berikut:

- (A) Menggunakan gelung **for**
(B) Kaedah rekursif tanpa menggunakan sebarang gelung

- (ii) Determine the time complexity of your function in Question 1(a)(i) above for the worst cases in terms of n. Justify your answers.

Tentukan kekompleksan masa fungsi anda dalam Soalan 1(a)(i) di atas bagi kes terburuk dalam sebutan n. Justifikasikan jawapan anda.

(30/100)

- (b) Given an implementation of the insertion sort algorithm below:

Diberi di bawah satu pelaksanaan algoritma isihan sisipan:

```
public void insertionsort(int[] data) {
    for (int i = 1, j; i < data.length; i++) {
        int tmp = data[i];
        for (j=i, j; j>0 && tmp < data[j-1];j--)
            data[j] = data[j-1];
        data[j] = tmp;
    }
}
```

- (i) The best case for this algorithm is when the data is already sorted. Why?

Kes terbaik bagi algoritma ini ialah apabila data sedia terisih. Mengapa?

- (ii) Insert counter statements into the codes above to count the number of moves and comparisons.

Sisipkan kenyataan-kenyataan pembilang ke dalam kod di atas untuk menghitung bilangan pergerakan dan perbandingan.

(30/100)

(c) Trace the following array using quicksort algorithm. Choose an appropriate pivot that will give:

- Best case
- Worst case running time.

For both cases give the running time in term of big Oh notation.

2 9 4 3 7 6 1

Surih algoritma isihan cepat apabila algoritma ini mengisi tatasusunan berikut dengan memilih pivot bersesuaian yang akan memberikan masa pelaksanaan:

- *Kes terbaik*
- *Kes terburuk*

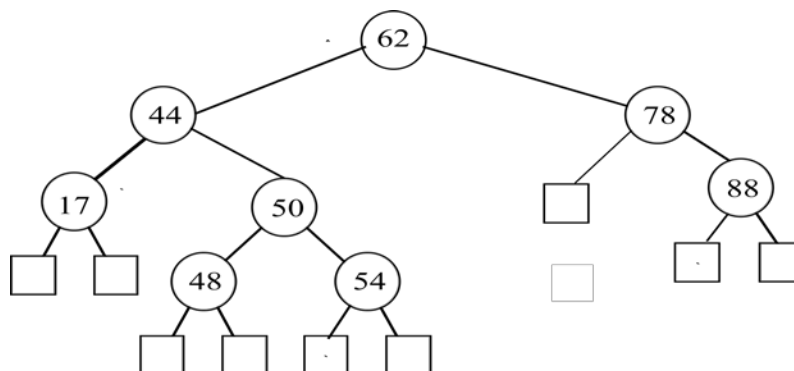
Bagi setiap kes, berikan masa pelaksanaan dalam bentuk tatatanda Oh besar.

2 9 4 3 7 6 1

(40/100)

2. (a) Given an AVL tree below.

Diberikan pepohon AVL di bawah.



(i) Insert the keys 52, 70 into the above tree. At any stage of the insertions, perform rotation(s) if necessary so that the tree is maintained as an AVL tree.

Sisipkan kunci 52, 70 ke dalam pepohon di atas. Pada mana-mana tahap penyisipan, lakukan putaran(-putaran) jika perlu supaya pepohon tersebut kekal sebagai pepohon AVL.

- (ii) Delete the key 62 from the (original) tree above. Perform rotation(s) if necessary so that the tree is maintained as an AVL tree.

Hapuskan kunci 62 daripada pepohon di atas (*pepohon yang asal*). Lakukan putaran(-putaran) jika perlu supaya pepohon tersebut kekal sebagai pepohon AVL.

(30/100)

- (b) Write the algorithm for `rotateRight()` function for the AVL Tree.

Tulis algoritma bagi fungsi `rotateRight()` bagi pepohon AVL.

(20/100)

- (c) Insert the following keys in the given order into a (2-4) tree that is initially empty: 5 16 22 45 2 10 18 30 50 12 1

Sisipkan kunci-kunci berikut dalam tertib yang diberikan ke dalam pepohon (2-4) yang pada awalnya kosong: 5 16 22 45 2 10 18 30 50 12 1

(30/100)

- (d) An airport is developing a computer simulation system of air-traffic control that handles events such as landing and takeoffs. Each event has a *time-stamp* that denotes the time when the event occurs. The simulation program needs to efficiently perform the following two fundamental operations:

- Insert an event with a given time-stamp (that is, add a future event).
- Extract the event with smallest time-stamp (that is, determine the next event to process).

What data structure should be used for the above operation? Why?

Satu lapangan terbang sedang membangunkan satu simulasi komputer bagi kawalan trafik udara yang boleh mengendalikan kejadian pendaratan dan pelepasan kapal terbang.. setiap kejadian mempunyai tanda masa yang menandakan masa bila kejadian tersebut berlaku. Atur cara simulasi ini perlu melaksanakan dua pengendalian asas berikut dengan berkesan:

- *Sisip kejadian dengan tanda masa yang diberi (iaitu menambahkan kejadian akan datang).*
- *Mengeluarkan kejadian yang mempunyai tanda masa paling kecil (iaitu menentukan kejadian seterusnya yang perlu diproses)*

Struktur data apa yang perlu digunakan bagi pengendalian di atas? Kenapa?

(20/100)

3. (a) Consider a tree T storing 100,000 entries. What is the height of T in worst case when

- (i) T is an AVL tree?
- (ii) T is a (2,4) tree?
- (iii) T is a binary search tree?

Pertimbangkan satu pepohon T menyimpan 100,000 kemasukan. Apakah ketinggian T bagi kes terburuk apabila

- (i) T adalah pepohon AVL?
- (ii) T adalah pepohon (2,4)?
- (iii) T adalah pepohon carian perduaan?

(30/100)

- (b) Give the resulting heap when starting with an empty heap, heapEnqueue is called consecutively for keys.

A L G O R I T H M S

- (i) Max heap
- (ii) Min heap

Berikan timbunan yang terhasil, bermula dengan timbunan kosong, heapEnqueue dipanggil bagi kunci-kunci.

A L G O R I T H M S

- (i) *Timbunan max*
- (ii) *Timbunan min*

(40/100)

- (c) Perform heapsort on the resulting heap (max) given in Question 2(b)(i) above.

Lakukan isihan timbun bagi timbunan(max) yang terhasil dalam Soalan 2(b)(i) di atas.

(30/100)

4. (a) Ali loves foreign languages and he wants to plan his course schedule for the following years. He is interested in the following nine language courses: LA15, LA16, LA22, LA31, LA32, LA126, LA127, LA141 and LA169. The course prerequisites are:
- LA15 : (none)
 - LA16 : LA15
 - LA22 : (none)
 - LA31 : LA15
 - LA32 : LA16, LA31
 - LA126 : LA22, LA32
 - LA127 : LA16
 - LA141 : LA22, LA16
 - LA169 : LA32

Draw an appropriate graph that reflects the order of courses he needs to take.

Find the sequence of courses that allows Ali to satisfy all the prerequisites.

Ali meminati bahasa asing dan ingin merancang skedul kursusnya bagi tahun-tahun berikut. Beliau berminat dengan sembilan kursus bahasa berikut: LA15, LA16, LA22, LA31, LA32, LA126, LA127, LA141 and LA169. Prasyarat untuk kursus-kursus ini adalah:

- *LA15 : (tiada)*
- *LA16 : LA15*
- *LA22 : (tiada)*
- *LA31 : LA15*
- *LA32 : LA16, LA31*
- *LA126 : LA22, LA32*
- *LA127 : LA16*
- *LA141 : LA22, LA16*
- *LA169 : LA32*

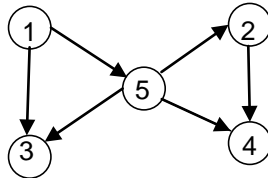
Lukis graf yang bersesuaian bagi menggambarkan turutan kursus yang beliau perlu ambil.

Cari jujukan kursus yang membenarkan Ali memenuhi segala prasyarat.

(40/100)

- (b) From the graph below, give **two (2)** different outputs that can be obtained for topological sort.

*Daripada graf di bawah, beri **dua (2)** output yang boleh diperoleh untuk isihan topologi.*



(20/100)

- (c) (i) Draw the 11-entry hash table that resulted from the hash function, $h(i) = (3i + 5) \bmod 11$, to hash the keys 12, 44, 13, 88, 23, 94, 11, 39, 20, 16 and 5, assuming collisions are handled by chaining.

Lukis jadual cincang 11-kemasukan yang terhasil daripada menggunakan fungsi cincang, $h(i) = (3i + 5) \bmod 11$, untuk mencincang kunci 12, 44, 13, 88, 23, 94, 11, 39, 20, 16 dan 5, mengandaikan perlanggaran dikendalikan oleh perantaraan.

- (ii) What is the result of the Question 4(c)(i) above, assuming collisions are handled by linear probing.

Apakah hasil dari Soalan 4(c)(i) di atas, jika mengandaikan perlanggaran dikendalikan oleh penjejakan linear.

(40/100)

5. (a) (i) Draw a standard trie for the following set of strings:

Lukis trie piawai bagi set rentetan berikut :

{ abab, baba, ccccc, bbaaaa, caa, bbaacc, cbcc, cbca }

- (ii) Draw a compressed trie for the set of strings given above.

Lukis trie termampat bagi set rentetan yang diberikan di atas.

(40/100)

- (b) Given a pseudocode for a simple string matching below:

Diberi satu pseudokod untuk satu pendekatan mudah bagi pemadanan rentetan di bawah:

```
bruteForceStringMatching(pattern P, text T)
  i = 0;
  while i ≤ |T| - |P|
    j = 0;
    while Ti == Pj and j < |P|
      i++;
      j++;
    if j == |P|
      return match at i - |P|;
    i = i - j + 1;
  return no match;
```

- (i) For the above algorithm, give the worst case complexity and a situation when the worst case occurs.

Bagi algoritma di atas beri kekompleksan kes terburuk dan satu situasi apabila kes terburuk berlaku.

- (ii) What is the number of comparisons executed in the algorithm above in terms of n and m where n is the length of the text and m the length of the pattern?

Apakah bilangan perbandingan yang dilakukan oleh algoritma di atas dalam sebutan n dan m ? Di sini n adalah panjang teks dan m adalah panjang pola.

- (iii) In order to analyse the number of comparisons of the algorithm above empirically we need to include counter statements to count the number of comparisons executed in the algorithm. Insert appropriate counter statements in the implementation above in order to carry out the task.

Untuk menganalisis bilangan perbandingan secara empirik algoritma di atas kita perlu memasukkan kenyataan pembilang untuk menghitung bilangan perbandingan yang dilakukan dalam algoritma berkenaan. Sisipkan kenyataan-kenyataan pembilang yang bersesuaian dalam pelaksanaan di atas untuk menjalankan tugas tersebut.

(40/100)

- (c) (i) Define a B-Tree of order m .

Takrifkan pepohon-B bertertib m .

- (ii) What is the main application of (a, b) trees in external memory?

Apakah kegunaan utama bagi pepohon (a, b) dalam ingatan luaran?

(20/100)